

Remarks

Claims 1 and 3-20 remain in this application. Claim 2 was previously canceled without prejudice. Claims 1, 10 and 15 are hereby amended. No new matter is being added.

Claim Rejections--Section 103

The pending claims were rejected under 35 U.S.C. § 103 as being unpatentable over Ayers in view of Prakash. This rejection is respectfully traversed with respect to the claims as now amended.

Amended claim 1 now recites as follows.

1. A method for cross-module in-lining, comprising:

in a first phase of a compiling process, the compiling process comprising a front-end phase, **an inter-procedural analysis phase in which cross-module analysis is performed on a plurality of modules, and a back-end phase in which the plurality of modules are processed individually**, the inter-procedural phase being the first phase,

determining to in-line a first function in a first module into a second function in a second module but not performing said in-line during the first phase;

providing the location of the first function;

providing instructions for in-lining to be performed in a second phase of the compiling process;

in the second phase of the compiling process, the back-end phase being the second phase,

following the instructions to in-line code of the first function into the second function in the second module without accessing the first module.

(Emphasis added.)

The claim now specifies that the first phase is the **inter-procedural phase** in which **cross-module analysis** is performed on a plurality of modules, while the second phase is the **back-end phase** in which the plurality of modules are **processed individually**. These phases are illustrated in FIG. 1 of the present application, which is reproduced below for convenience of reference.

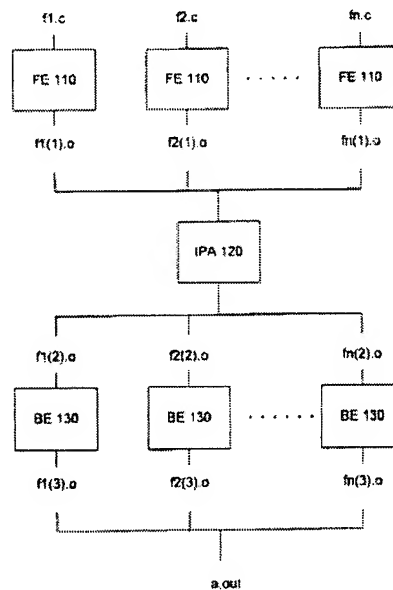


FIG. 1

100

Above FIG. 1 shows the inter-procedural analysis phase (IPA 120) which performs **cross-module** analysis on a plurality of modules and the back-end phase (BE 130) in which the modules are processed **individually**.

As further specified in the amended claim 1, the **in-lining determination** is made in the first (inter-procedural analysis) phase, while the **actual in-lining** is performed in the second (back-end) phase. This is a unique and inventive aspect of the claimed invention.

The thinking behind this unique and inventive aspect is that, while in-lining is conventionally performed earlier and more naturally in the IPA phase (as the IPA 120

has access to all the modules), the IPA phase is not parallelizable or not easily parallelizable. In other words, the IPA phase runs in a **serial** fashion and can become a compile-time bottleneck for large programs. In contrast, the FE and BE phases are parallelizable. Hence, applicants have invented a technique where the in-lining **decisions** are made in the IPA phase, but the actual in-lining is **delayed** until the BE phase. This advantageously allows the actual in-lining to be performed in a **parallel** manner during the BE phase.

While the claimed invention focuses on in-lining and the determination and performance thereof, Prakash's disclosure relates to inter-procedural optimization. In particular, Prakash discloses, in the event of user changes in a subset of the source files, how to incrementally perform inter-procedural optimization without re-doing everything from scratch.

The figure from Prakash which corresponds to FIG. 1 of the present application appears to be FIG. 5. For convenience of reference, FIG. 5 of Prakash is reproduced below.

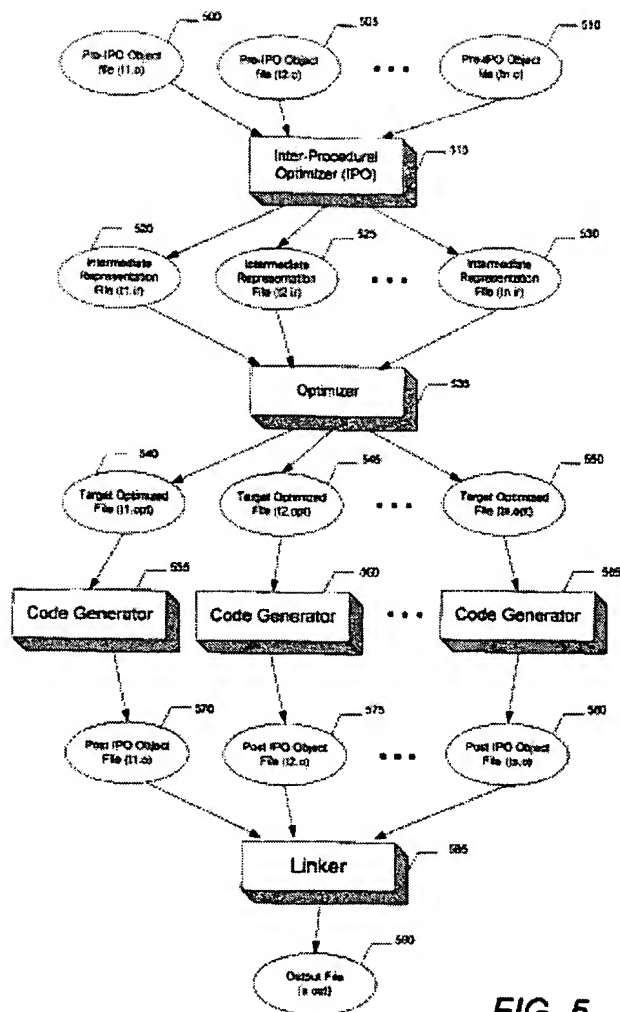


FIG. 5

As seen above, the inter-procedural optimizer (IPO) and the Optimizer in Prakash perform **cross-module** analysis, while the Code Generators process the modules **individually**. Hence, the IPO and Optimizer appears to be similar to the first (inter-procedural analysis) phase of claim 1, while the Code Generators appears to be similar to the second (back-end) phase of claim 1.

In regards to in-lining, paragraph [0059] of Prakash merely recites as follows. "For example, if a PostIPO object file inlines a function 'foo' from a first file, a second file assumes a certain definition of 'foo.' ... Since the first file **had already inlined** the previous version of the function 'foo,' in a naïve implementation of this feature the resulting executable file will be incorrect." (Emphasis added.) In other words, Prakash merely discloses that inlining of a function may **already exist** in a PostIPO object file. In regards to where the in-lining was **actually performed**, Prakash

implies that **the in-lining (along with other cross module optimizations) is performed by the IPO and Optimizer (as is conventionally done)**. For example, paragraph [0039] of Prakash states, "... the IPO ... passes 'ir' information to the optimizer 535 in a single invocation step whereupon interprocedural optimization is performed."

Therefore, applicants respectfully submit that the addition of Prakash to Ayers does not disclose or teach the claimed invention per amended claim 1. Hence, applicant respectfully submits that amended claim 1 now overcomes this rejection.

Claims 3-9 depend from claim 1 and so are also distinguished over Ayers and Prakash for at least the reasons discussed above.

Furthermore, claim 4 (with intervening claim 3) further recites an additional function (the "third" function) to contain a copy of the function to be in-lined, and "getting rid of the third function in the module ... after using that third function to in-line its code" No such disclosure or teaching is found in Ayers or Prakash. Hence, applicants respectfully further submit that claim 4 is further distinguished over Ayers and Prakash.

Claims 10 and 15 are amended similarly as claim 1. Hence, for at least the reasons discussed above in relation to claim 1, applicants respectfully submit that claims 10 and 15 now also overcome this rejection.

Claims 11-14 and claims 16-20 depend from claims 10 and 15, respectively. Hence, claims 11-14 and 16-20 also overcome this rejection.

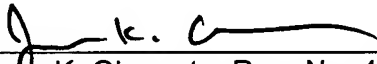
Conclusion


For the above-discussed reasons, applicant respectfully submits that the pending claims are now patentably distinguished over the cited art. Favorable action is respectfully requested.

The Examiner is also invited to call the below-referenced attorney to discuss this case.

Respectfully Submitted,

Dated: October 29, 2007


James K. Okamoto, Reg. No. 40,110
Tel: (408) 436-2111
Fax: (408) 436-2114

CERTIFICATE OF MAILING			
I hereby certify that this correspondence, including the enclosures identified herein, is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below. If the Express Mail Mailing Number is filled in below, then this correspondence is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service pursuant to 37 CFR 1.10.			
Signature:			
Typed or Printed Name:	James K. Okamoto	Dated:	October 29, 2007
Express Mail Mailing Number (optional):			